(54) **Pixellated to vector image conversion**

(57) An electronic imaging system receives a pixellated image and stores it in a store (14). An edge information deriving unit (18) produces data relating to edges in the image and stores this in another store (20). An expander (26) takes the stored edge information and expands it. Each edge in the image is then "double-lined" by a unit (30) prior to sets of co-ordinates representing the edge information being generated and stored in a co-ordinate store (4). A vector fit unit (8) then produces vectors corresponding to edges in the image and these are stored in vector store (10).

Fig.7.
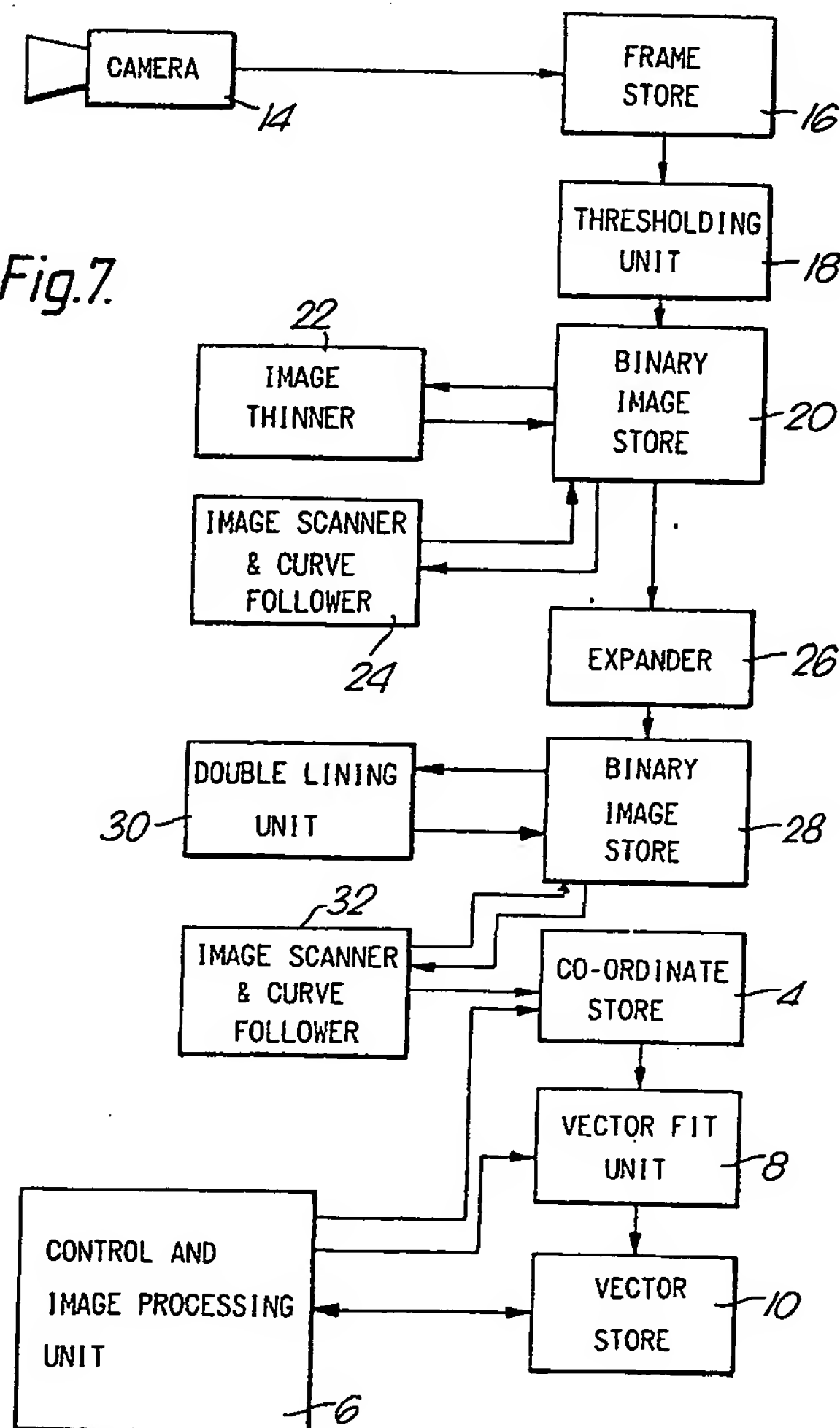
GB 2 263 215 A

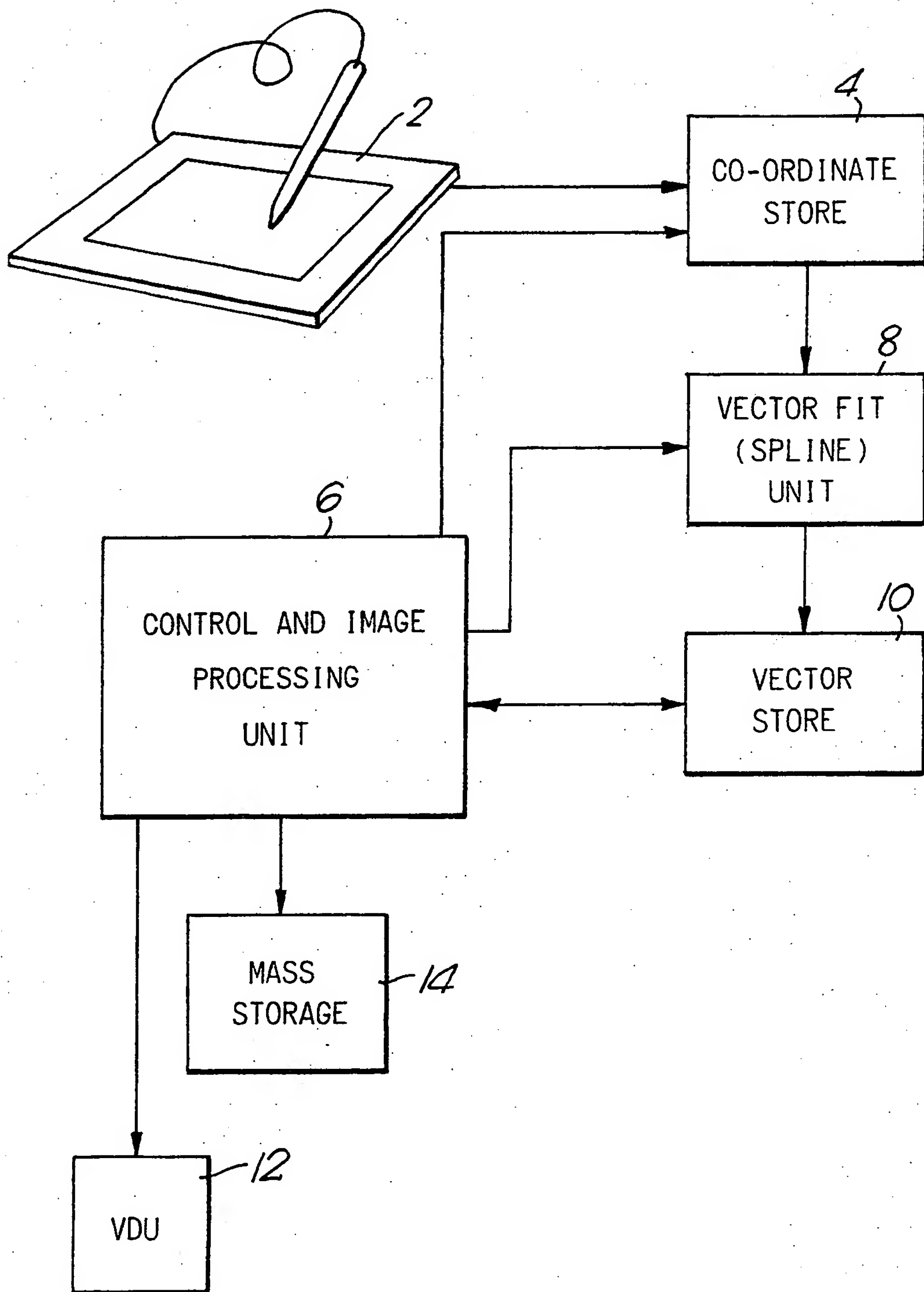# Fig.1.

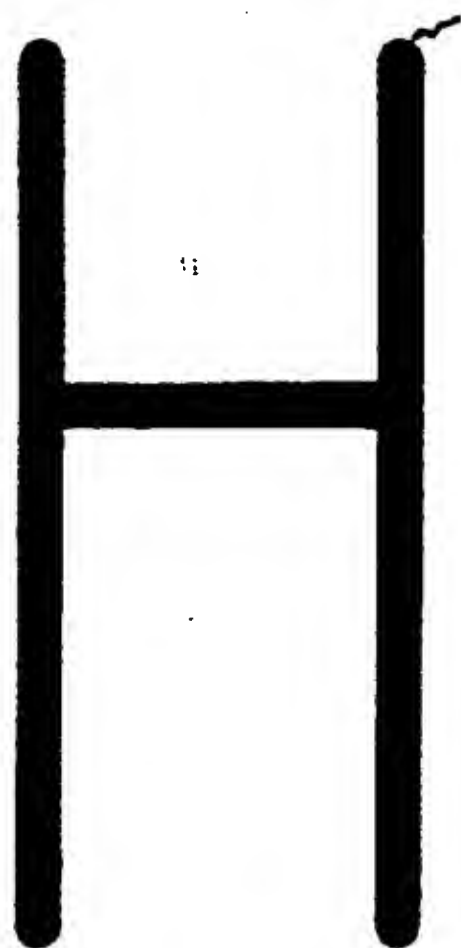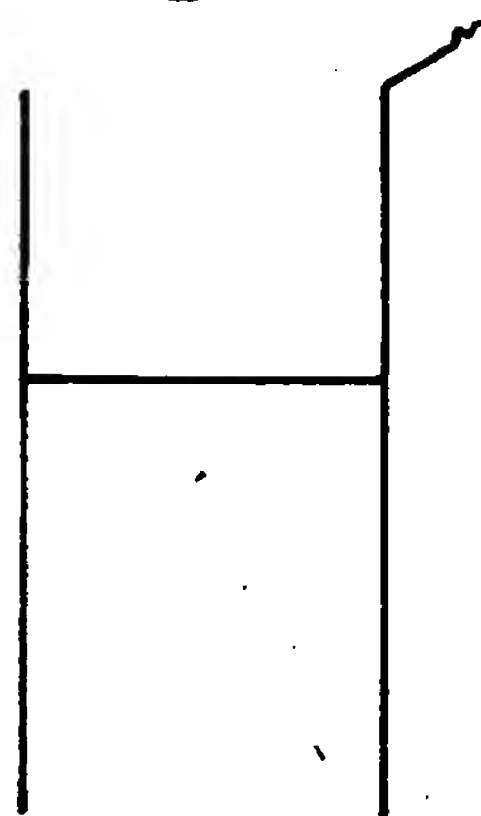# Fig.2a.

OBJECT WITH NOISE

Fig.2b.

OBJECT AFTER THINNING SHOWING
SPURIOUS LINE

Fig.3.

POINT 1

POINT 3

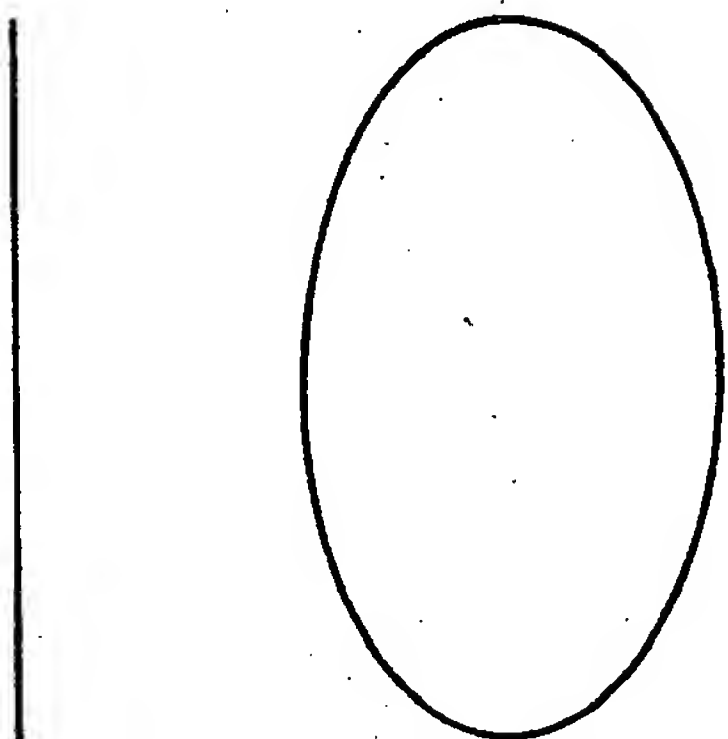POINT 2

POINT 4
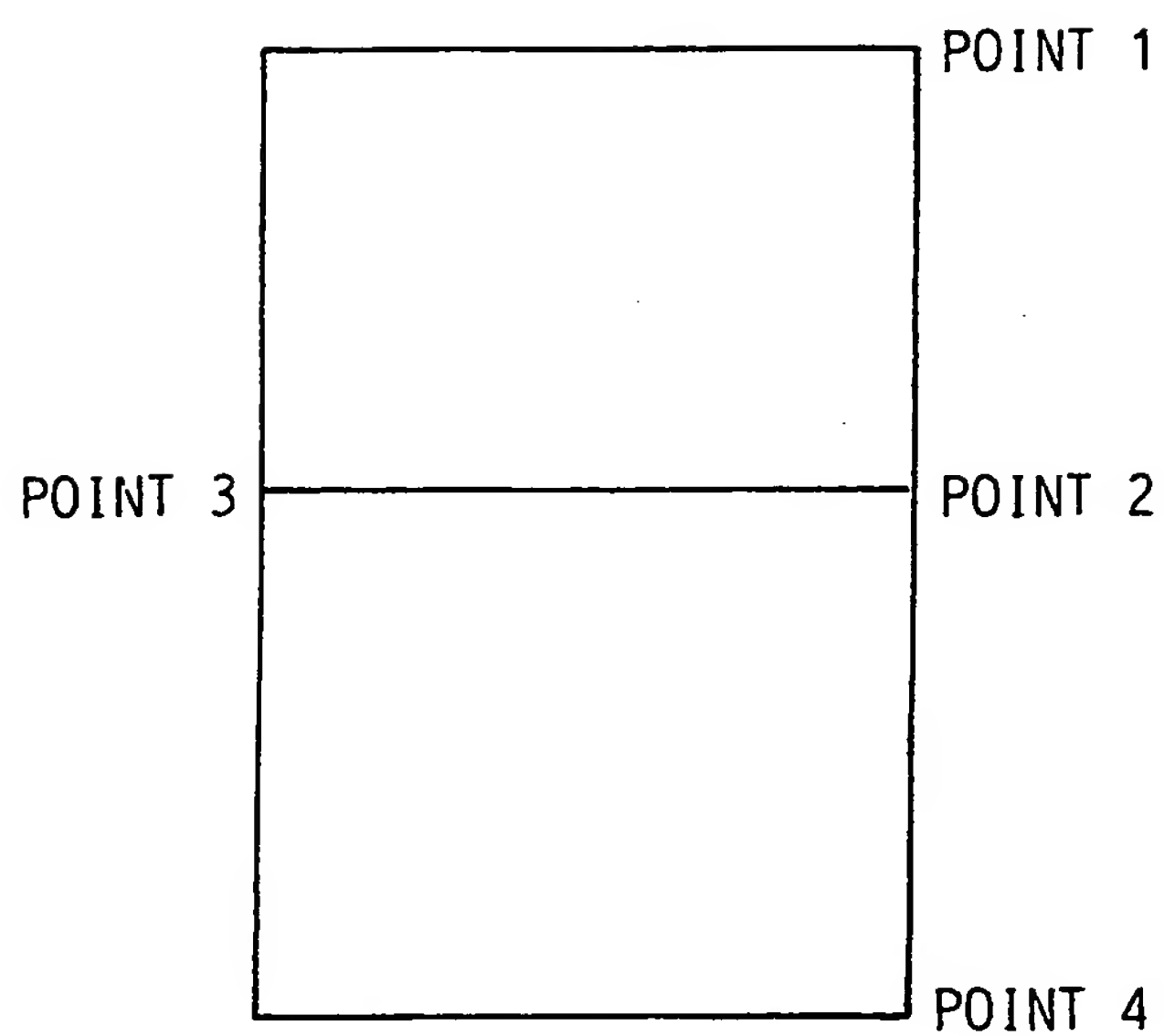
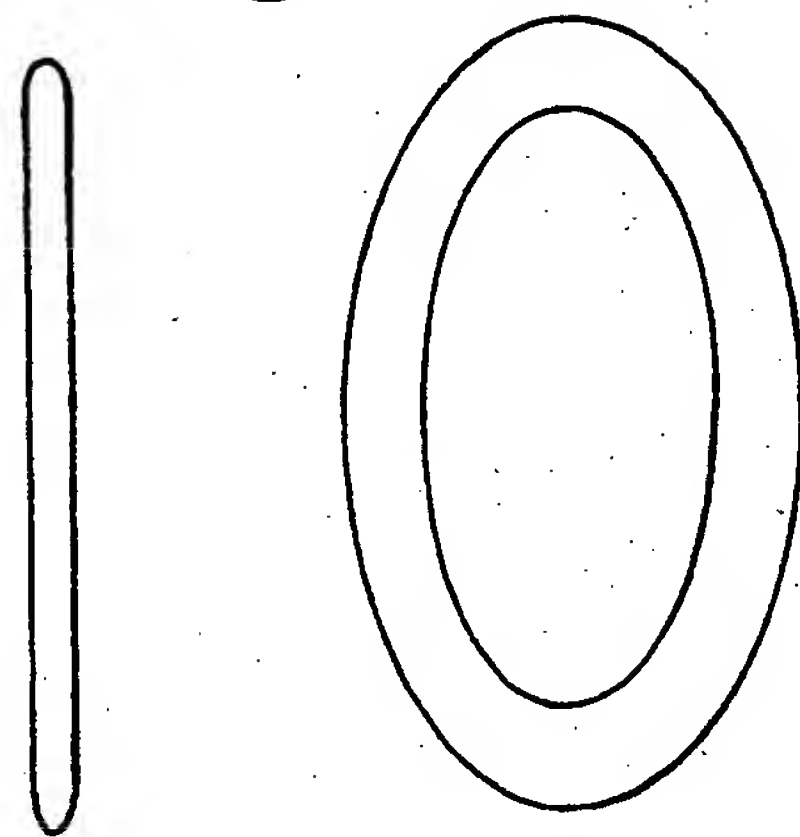OBJECT WITH JUCTIONS

Fig.4.
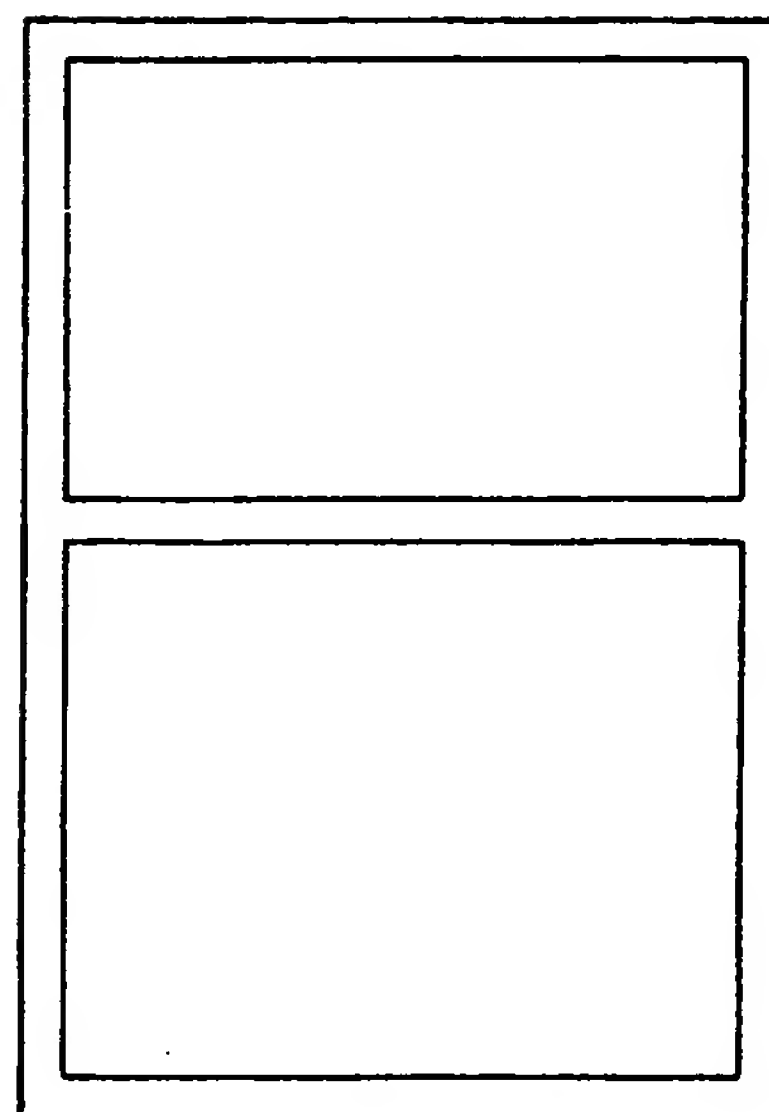
OBJECT IN FIGURE 3 AFTER
DOUBLE LINING

Fig.5a.

SAMPLE OBJECTS
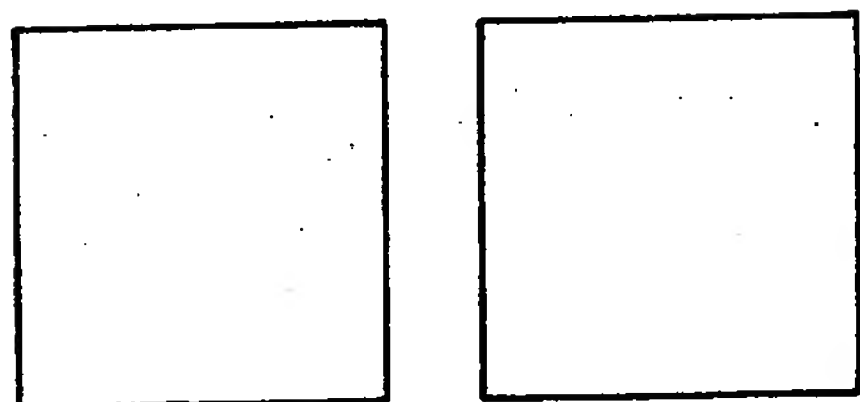
Fig.5b.

EFFECT OF DOUBLE LINING

Fig.6a.

ORIGINALS

Fig.6b.

PROCESSED OBJECTS
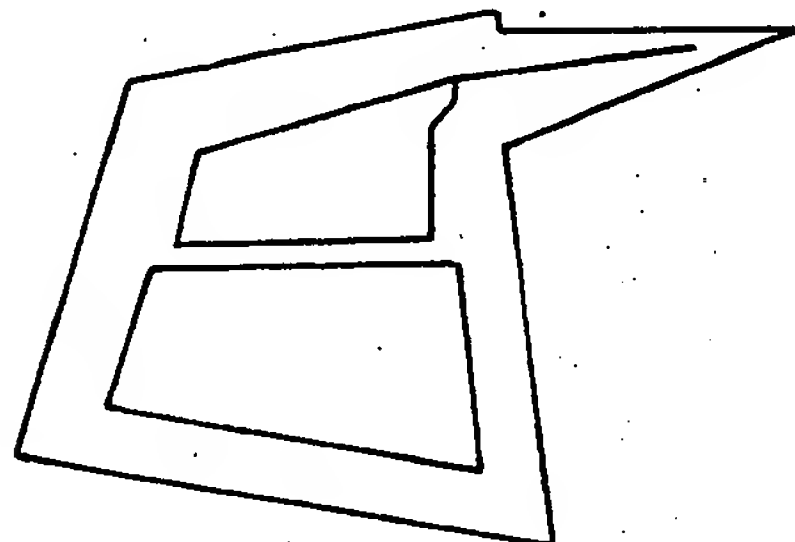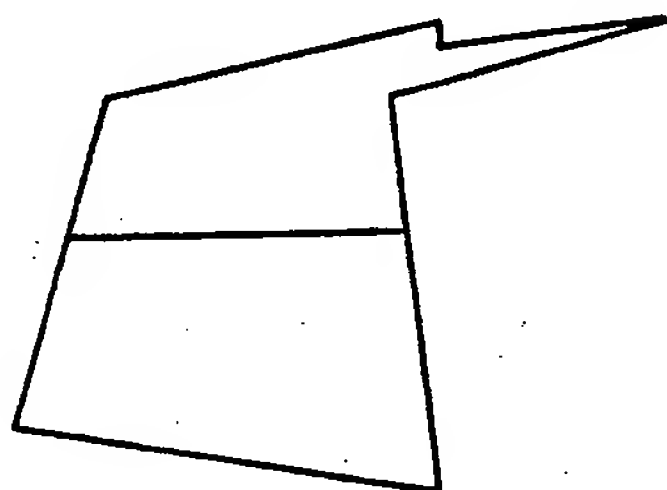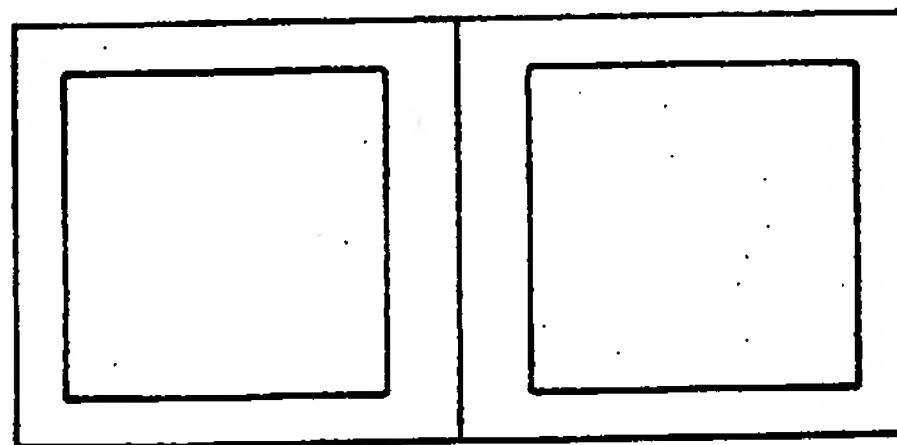
CREATION OF CONNECTED OBJECTS AND LINES BY DOUBLE LINING

Fig.7.
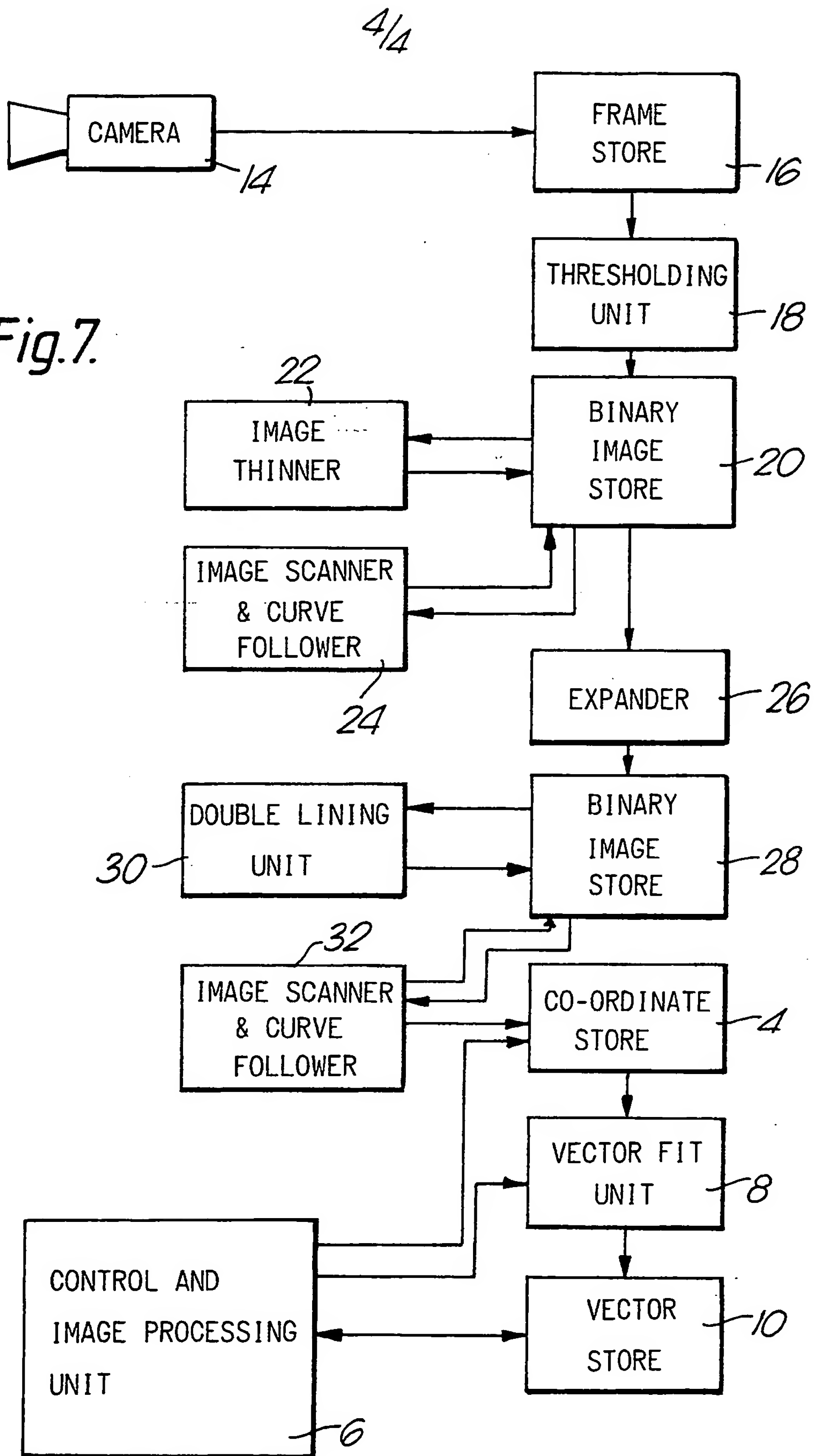
ELECTRONIC IMAGING SYSTEMS

This invention relates to electronic imaging systems of the type which represent images as a set of vectors.

Pixel based imaging systems for techniques such as electronic painting and animation are already known. It has been found that systems which store images as sets of vectors rather than as a grid of pixels are particularly versatile and efficient.

Images are usually read into such a system using a tablet and stylus. An artist draws with the stylus on the tablet and this sends a set of co-ordinate values, representing the position and past movement of the pen, from the tablet to the imaging system. The system then converts this string of co-ordinates to a vector by fitting a spline through them.

Once these vectors representing lines have been stored in the system a user can assign any width, colour, texture, etc he wishes to the line for the purposes of display. Similarly, any areas around which a vector, or set of vectors, defines an outline can be infilled as the user desires.

A vector based imaging system of the type summarised above can best read images from a tablet and stylus. However, it is desirable for the imaging system to be able to receive other images such as pixel based images produced, for example, from hand drawn images or from video images. Images of this type pose severe problems for vectorisation since they are represented by shading values applied to pixels rather than by co-ordinate values.

Preferred embodiments of the present invention provide systems which can produce a vectorised image from a pixel based image.

The invention is defined in its various aspects in the

appended claims to which reference should now be made.

An embodiment of the invention will now be described in detail by way of example with reference to the accompanying drawings in which:

Figure 1 shows a block diagram of an image vectorisation system arranged to receive images drawn in a graphics tablet;

Figure 2 a) and b) respectively show an image before and after line thinning by an embodiment of the invention;

Figure 3 shows a line thinned object with junctions;

Figure 4 shows the object of figure 3 after double lining in accordance with an embodiment of the invention;

Figure 5 a) and b) respectively show the effect of double lining on simple objects;

Figure 6 shows some of the problems which can occur with double-lining on complex objects; and

Figure 7 shows a block diagram of a system embodying the invention.

The block diagram of the image vectorisation system shown in figure 1 comprises a tablet and stylus 2 which can be operated by a user and which will produce co-ordinate values in dependence on the position of the stylus on the tablet. As the stylus is moved on the tablet a continuous stream of co-ordinate values will be produced. These co-ordinate values are read out from the tablet 2 to a co-ordinate store 4. This is operated, under the control of a control and image processing unit 6, to store sets of co-ordinates corresponding to lines drawn with the tablet and stylus 2.

A vector fitting unit 8 is also under the control of the control and image processing unit 6. Its purpose is to perform a spline fitting routine on a set of co-ordinates. It is controlled to receive a set of co-ordinates from the co-ordinate store 4 on which it can then perform its spline fitting routine. The resultant spline or vector represents the line defined by the set of co-ordinates supplied to the unit 8.

The resultant vector is read out to a vector store 10 which can be accessed by the control and image processing unit 6.

The vector store 10 is large enough to store all the vectors needed to represent an image and these can be accessed by the control and image processing unit 6 in order to modify that image whilst displaying it on a VDU 12. Once the form of an image has been decided upon it can be permanently stored as a set of vectors with data relating to the display of each vector in a mass storage device 14 coupled to the control and image processing unit 6.

The control and image processing unit 6 has a plurality of user inputs (not illustrated) which can be used to assign characteristics to the lines represented by the vectors in the store 10 for display. Functions such as infilling routines and other standard graphics routines are also available.

Embodiments of the present invention enable a set of co-ordinates to be generated from a pixellated image and fed into the co-ordinate store 4. The vector fitting can then be performed on them in the vector fit unit 8 and any vectors produced stored in vector store 10. These can then be used by the control and image processing unit 6 to manipulate the images represented by the vectors.

In order to produce a set of co-ordinates from a pixellated image it is necessary to perform a series of pre-processing steps on the image. The embodiment described herein shows how a line

drawing of black ink on white paper can be input to an imaging system of the type shown in figure 1 and stored in vectorised form.

The image can initially be captured and converted to pixellated form by use of a video camera and a framestore.

Once a pixellated version of the image has been produced in a framestore it is then necessary to formulate a binary version of the image in which each pixel is assigned either to the background or to a line. One way of doing this is to extract the luminance component of the original image (assuming it was originally scanned in RGB) format.

This will produce an 8-bit grey scale image. Assuming there was even illumination of the image when it was scanned by the video camera it should be possible to choose a simple threshold value and use this to assign pixels to the background or to the line.

If the image was originally illuminated unevenly some of the "white" background pixels could in fact be darker than some of the "black" line pixels. This makes it impossible, in practice, to choose a simple threshold value to partition the image into background and line pixels.

A pre-processing algorithm for grey scale images can be used to overcome the problem caused by uneven illumination. an adaptive technique is described in the following process and this has been found to give very goods results. The steps of the process are as follows:

1. The local average of the luminance value for the current pixel is calculated from an N by N pixel region surrounding the current pixel.

2. The local average derived in step 1 is substracted from the current pixel value to produce a difference value for that pixel, delta.

3. If the difference value delta for the current pixel is greater than 0 then the output for that value is set to 255. If delta is less than 0 then the output for that pixel is set to 255 + delta. In the latter case this leads to a value less than 255 since delta is negative.

4. A global threshold value is now assigned to the whole image to produce the required binary image which has logical 1's ones representing pixels on lines and logical 0's noughts representing pixels in the background.

The global threshold used in step 4 can be determined in any desired way. For example, the distribution of grey scale values throughout the image could be examined and a threshold selected in accordance with the distribution. Alternatively the average of the optimum thresholds for a set of typical images previously input to the system could be used.

The thresholded image can be stored by using one bit for each pixel and setting this to a logical 0 or logical 1 in accordance with the result of the thresholding process. This gives the binary image. The lines in this image will be of varying thickness and a number of spurious points may be present. The next stage in the production of a set of co-ordinate values for vectorisation is to remove these points and to "thin" all the lines in the image so that they are only one pixel wide.

An iterative thinning algorithm is known from Zahng and Suen Comm. ACM, Vol, No. 3, pp. 236-239, " A Fast Parallel Algorithm for Thinning Digital Patterns". This is also described in Gonzales and Wintz, pp. 399-402. Using such an algorithm gives good results. The only problem that has been encountered is that the uneven thickness of some lines can occasionally cause the appearance of spurious lines in the output image. Figure 2 shows an example of

how these can arise. This shows a binary representation of the letter H with some noise at the top right hand corner of the letter. After thinning this noise is amplified and in fact causes the appearance of an extra line. The amplification occurs since the thinning process tends to shrink the size of objects.

To overcome this type of problem a modified version of the thinning algorithm has been used. This involves relaxing the conditions required for an image point to be deleted from the image after the first few passes thereby causing noisy points to be removed. The process then reverts to the original form.

Once the image has been thinned it is in a suitable form to be used to produce a set of co-ordinates for vectorisation. The image consists of a series of one pixel wide lines and these can be followed to produce the required sets of co-ordinates. There are many possible curve following routines which can be used to define the sets of co-ordinates and one of these is as follows:

1.   The image is scanned, line, by line until a black pixel, representing a line, is found.

2.   This black pixel is assigned as the current pixel and is marked as a line point. This is done by setting another bit corresponding to that pixel.

3.   The 8 pixels surrounding the current pixel are examined.

4.   If none of these surrounding pixels are unmarked line points, the line is finished and the algorithm jumps to step 7.

5.   If one of the neighbouring pixels is an unmarked line point then this is selected as the next pixel in the sequence.

6.   Return to step 3.

7.   Continue from step 1 until the end of the image is reached.

This approach will work for unconnected lines, curves, and shapes. A problem arises when lines or objects intersect.

If we consider the object in figure 3 and a line is followed from the pixel marked point 1 to the pixel marked point 2, a decision must then be taken as to whether the line leading to point 3 or point 4 should be followed. The curve following routine only has information from surrounding pixels on which to base a decision and this is insufficient to always make the correct decision. For example, the image of figure 3 could be a large rectangle with a dividing line or two adjacent squares, or even one large rectangle enclosing two smaller squares. If the second or third of the above examples is true then the pixels in the middle line must be used twice, one for each square.

A particularly advantageous way to analyse the scene of figure 3 is the example in which there is one large rectangle enclosing two smaller squares. When vectorising such an image the approach would be to provide a vector describing the overall outline shape and one for all the constituent objects. This can be achieved if every line were replaced by two new lines one an "external" defining the external outline and the other an "internal" line defining each shape within the external outline. This process is known as double lining and in order to implement it on a binary image the following steps are performed:

1.   Every white pixel in the image that has at least one neighbour that is black is replaced with a black pixel.

2.   Every black pixel in the image is replace with a white pixel.

If this is done to the object of figure 3 the result is the object of figure 4. This shows three distinct closed loops describing the three component parts of the image. Each of these

loops can be followed unambiguously (no junctions are met) to produce a set of co-ordinates for which vectors can then be derived.

One problem with this double lining process is the fact that single lines and closed loops are replaced with two versions of themselves as shown in figure 5b. A solution to this is to scan the thinned image to remove single lines and closed unconnected loops prior to applying the double lining technique to the remainder of the image. The closed loops and single line can be used as direct inputs to the curve following process and can hence be vectorised.

A more serious problem with the double lining technique is that if two unconnected objects are very close together e.g less than three pixels, their outlines will overlap after the double lining process. This may produce a curve which has junctions of the type illustrated in figure 3 and hence has the same problem since junctions will then be encountered when a curve is followed.

The problem is overcome by ensuring that lines are never closer together than two pixels or smaller in width than two pixels. This is achieved by expanding the size of the image by a factor of 3 in both horizontal and vertical directions.

This expanded image can then be double lined and the resulting curves will not overlap. The curves can then be followed to produce a set of co-ordinates. These co-ordinates can then be divided by a factor of three to map the co-ordinates onto the original image plane. Another problem with double lining is that curves of different size are produced for internal and external shapes. Adjoining shapes therefore become separated by one pixel. This can be avoided by adding a one to the x and y values of the co-ordinates derived for each curve prior to division by three. This will cause all the curves to fit together.

The sets of x and y co-ordinates defining each curve or

outline in the image can then be vectorised.

Putting all the above processes together produces an overall process for producing a set of co-ordinates for input to a vectorisation based imaging system from a black and white line drawing. The steps of the process are as follows:

1. Pre-process the image by thresholding or whatever other process is appropriate to derive a binary image.

2. Thin the image to produce a skeletonised version with all lines one pixel wide.

3. Scan the skeletonised image and the mark all the end points of lines and any junctions between lines.

4. From each end point follow the line until another end point or a junction is reached. The co-ordinates are stored and those lines already encountered removed from the image.

5. Repeat from step 3 until no more end points are found. This is because removing lines (step 4) can create new lines and can also remove existing junctions.

6. For all junctions remaining in the image follow the lines connecting them to other junctions marking all the points in between.

After this process the only unmarked line pixels still in the image should belong to closed unconnected loops. The next stage is to vectorise these using the following steps:

7. Scan the image for unmarked line points then follow the line storing and removing the pixels in the line from the image as in step 4.

8.   Repeat step 7 until no umarked points are left.

The image should now consist of the previously marked points connected to junctions. The image is now expanded and double lined using the following steps:

9.   Expand the image by a factor of 3 in the x and y directions.

10.   Double line the image.

11.   Scan the image for line pixels and follow them as in step 7 then increase co-ordinate values by one prior to division by 3.

12.   Continue until no line pixels are left unmarked.

The expansion of the image by three in both directions necessitates an increase in storage capacity by a factor of 9. However, the image is a binary one, i.e. only one bit per pixel is required and the storage space required for the increased image is therefore less than that needed for an unexpanded 24 bit digital image.

A block diagram of a system which will implement the above process is illustrated in figure 7. This comprises the control and image processing unit 6, the co-ordinate store 4, the ventor fit unit 8, and the vector store 10 shown in figure 1.   The mass storage device 14 and the VDU 12 of figure 1 will also be present, but, for the purposes of clarity, these are not shown in figure 7.

A camera 14 is used to take pictures of an image to be read into the system and a picture is stored in a framestore 16. A thresholding unit 18 receives data from the from the framestore 16 and from this produces a binary image which is then output to a binary image store 20.

An image thinner 22 operates on the binary image in store 20

to produce a skeletonised image of the type shown in figure 2b. An image scanner and curve follower 24 then operates on the thinned image to find all the end points and junctions of lines. This image scanner and curve follower removes individual lines and closed unconnected loops from the image before any further processing can take place. Sets of co-ordinates representing these lines and closed unconnected loops are fed back to the control and image processing unit and from there to the co-ordinate store 4. For the purposes of clarity this connection is not shown.

Once all the unconnected lines and closed unconnected loops have been removed from in store 20 the image it is expanded, typically by a factor of 3, in the x and y directions, in an expander 26. The expanded binary image is then stored in a second binary image store 28. Connected to this binary image store 28 is a double lining unit 30 which operates on the data stored in the store 28 to produce an image of the type shown in figure 4. An image scanner and curve follower 32 then operates on this double lined image to produce sets of co-ordinates which are stored in co-ordinate store 4. The rest of the apparatus operate in the same manner as that shown in figure 1.

The whole operation of figure 7 is of course under the control of the control and image processing unit 6 but connections between this and the various units of the apparatus are, for the purposes of clarity, not shown.

The various units of figure 7, for example, the image thinner 22 and the double lining unit 30, can be implemented in hardware using straighforward techniques. Alternatively the functions performed by the various units could all be performed in software.

Apparatus of the type shown in figure 7 gives good results for a wide range of black and white art work, both line drawing and block based. It is even capable of processing continuous tone images such as computer generated logos providing there is

sufficient contrast in tone between adjacent blocks.

It has been found that if the image is initially very noisy then a pre-processing step of median filtering will improve the performance to a significant degree.

In cases where the input image consists of black and white art work, evenly illuminated for the camera 14 then it is possible to omit the thresholding unit 18 and, in some cases, the image thinning. Thus a simple binary image would be derived from framestore 16 and this would form the input to expander 26. Colour images can be processed in a similar manner with the use of a segmentation technique to transform the image into an edge map of the colour blocks of which it is comprised. This is of course more complex than the binary image in the example described above.

In a complete imaging system a user would be presented with a number of vectorisation options based on the type of input via the camera 14. Typical classes might be: line drawings, black and white art work, colour graphics, and continuous tone images. A useful system will give the operator the option of intervening in the processing of the image at certain stages such as before line thinning to allow the image to be retouched or to allow special effects manipulation of the image. Thus a user will be able to retouch the image using a tablet and stylus to modify either the image stored in framestore 16 or that stored in binary image store 20.

Using the present invention enables any image to be used as an input to a vector based imaging system without having to draw on a graphics tablet with a stylus. The system is therefore attractive to people who do not wish to use the graphics tablet.

CLAIMS

1.   An electronic imaging system comprising means for receiving and storing a pixellated image, means for deriving edge information from the pixellated image, means for storing the edge information in pixellated form, means for expanding the stored image to cover a greater number of pixels, double-line deriving means responsive to stored edge information to derive a pair of edges from each stored edge, means for scanning the output of the double-line deriving means to produce sets of co-ordinates representing edges in the pixellated image, means for fitting a vector to each set of co-ordinates, and means for storing each resultant vector.

2.   An electronic imaging system according to claim 1 in which the pixellated image represents a two-tone line drawing.

3.   An electronic imaging system according to claim 2 in which the edge information deriving means first dervies a binary representation of the two-tone line drawing.

4.   An electronic imaging system according to claim 3 including image thinning means for reducing the width of stored lines in the binary representation to one pixel.

5.   An electronic imaging system according to claim 1 in which the expanding means expands by a factor of 3.

6.   An electronic imaging system according to claims 1, 2, or 3 in which edge information is stored as lines one pixel wide.

7.   An electronic imaging system according to claim 4 or 6 including means for producing a pair of adjacent lines from each stored line.

8.   An electronic imaging system according to claim 7 including means for detecting whether a pixel is on a line, and means for

removing line pixels from the image and denoting each pixel
adjacent to a removed pixel as a new line pixel.

9.   A electronic imaging system according to any preceding claim
including means to remove any unconnected edges or closed loops
from the stored edge information and means to provide a set of
co-ordinates representing each removed edge and closed loop to the
vector fitting means.

10.   An electronic imaging system according to any preceding claim
in which the pixellated image is provided by a video camera.

11.   An electronic imaging system according to any preceding claim
including a user contol to designate the type of the pixellated
image.

12.   An electronic imaging system substantially as herein
described with reference to the accompanying drawings.

Patents Act 1977
Examiner's report to the Comptroller under Section 17 (The Search Report)

Application number

9200131.2

Relevant Technical fields

(i) UK Cl (Edition   K   ) H4T (TBBA, TBBD, TBBX, TBCX, TBEX, TDCD); H4F (FGXX)

(ii) Int CL (Edition   5   ) G06F

Databases (see over)

(i) UK Patent Office

(ii)   ONLINE DATABASES: WPI

Search Examiner

R F KING

Date of Search

14 MAY 1992

Documents considered relevant following a search in respect of claims   1-11

Category (see over)	Identity of document and relevant passages	Relevant to claim(s)
	NONE	1

SF2(p)

HD - c:\wp51\doc99\fil000296

Category	Identity of document and relevant passages	Relevant to claim(s)

BEST AVAILABLE COPY

Categories of documents

X: Document indicating lack of novelty or of inventive step.

Y: Document indicating lack of inventive step if combined with one or more other documents of the same category.

A: Document indicating technological background and/or state of the art.

P: Document published on or after the declared priority date but before the filing date of the present application.

E: Patent document published on or after, but with priority date earlier than, the filing date of the present application.

&: Member of the same patent family, corresponding document.

Databases: The UK Patent Office database comprises classified collections of GB, EP, WO and US patent specifications as outlined periodically in the Official Journal (Patents). The on-line databases considered for search are also listed periodically in the Official Journal (Patents).